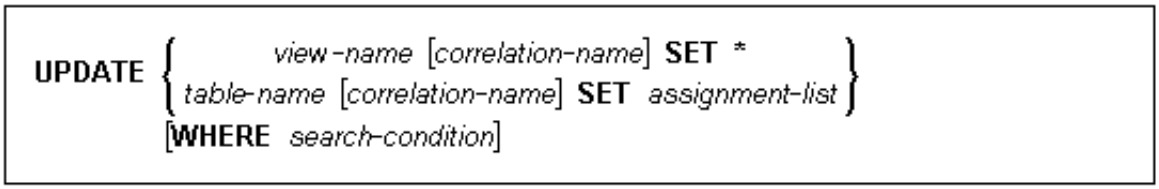
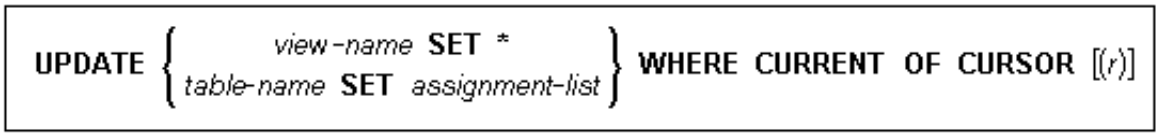
UPDATE

Syntax 1 - Searched UPDATE



Syntax 2 - Positioned UPDATE



Function

The SQL UPDATE statement is used to perform an UPDATE operation on either rows in a table without using a cursor ("searched" UPDATE) or columns in a row to which a cursor is positioned ("positioned" UPDATE).

The "searched" UPDATE statement is a stand-alone statement not related to any SELECT statement. With a single statement you can update zero, one, multiple or all rows of a table. The rows to be updated are determined by a *search-condition* that is applied to the table. Optionally, view and table names can be assigned a *correlation-name*.

The "positioned" UPDATE statement always refers to a cursor within a database loop. Thus, the table or view referenced by a positioned UPDATE statement must be the same as the one referenced by the corresponding SELECT statement; otherwise an error message is returned. A positioned UPDATE cannot be used with a non-cursor selection.

See Basic Syntactical Items for further information on *view-name*, *table-name*, *authorization-identifier* and *correlation-name.*

Note:
The number of rows that have actually been updated with a "searched" UPDATE can be ascertained by using the system variable *ROWCOUNT in the Natural Reference documentation.

SET Clause

If a view has been specified for updating, an asterisk (*) has to be specified in the SET clause, because all columns of the view must be updated.

If a table has been specified for updating, the SET clause must contain either an assignment-list or the name of view which contains the columns to be updated.

assignment-list

In an *assignment-list*, you can assign values to one or more columns. A value can be either a *scalar-expression* or NULL.

See further information on Scalar Expressions.

If the value NULL has been assigned, it means that the addressed field is to contain no value (not even the value "0" or "blank").

WHERE search-condition

The WHERE clause is used to specify the selection criteria for the rows to be updated.

If no WHERE clause is specified, the entire table is updated.

Statement Reference - r

The "(*r*)" notation is used to reference the statement which was used to select the row to be updated. If no statement reference is specified, the UPDATE statement is related to the innermost active processing loop in which a database record was selected.

Examples

Example of Searched UPDATE:

```
DEFINE DATA LOCAL
1 PERS VIEW OF SQL-PERSONNEL
2 NAME
2 AGE
...
END-DEFINE
...
ASSIGN AGE = 45
ASSIGN NAME = 'SCHMIDT'
UPDATE PERS SET * WHERE NAME = 'SCHMIDT'
...
```

Example of Searched UPDATE with *assignment-list*:

```
DEFINE DATA LOCAL
1 PERS VIEW OF SQL-PERSONNEL
2 NAME
2 AGE
...
END-DEFINE
...
UPDATE SQL-PERSONNEL SET AGE = AGE + 1 WHERE NAME = 'SCHMIDT'
...
```


Example of Positioned UPDATE:

```
DEFINE DATA LOCAL
1 PERS VIEW OF SQL-PERSONNEL
2 NAME
2 AGE
...
END-DEFINE
...
SELECT * INTO PERS FROM SQL_PERSONNEL WHERE NAME = 'SCHMIDT'
COMPUTE AGE = AGE + 1
UPDATE PERS SET * WHERE CURRENT OF CURSOR
END-SELECT
...
```

Example of Positioned UPDATE with *assignment-list*:

```
DEFINE DATA LOCAL
1 PERS VIEW OF SQL-PERSONNEL
2 NAME
2 AGE
...
END-DEFINE
...
SELECT * INTO PERS FROM SQL-PERSONNEL WHERE NAME = 'SCHMIDT'
UPDATE SQL-PERSONNEL SET AGE = AGE + 1 WHERE CURRENT OF CURSOR
END-SELECT
...
```

